スーパーテクニカルサーバSR11000モデルJ1のノードアーキテクチャと性能評価

青木秀貴[†] 中村友洋[†] 助川直伸[†]
齋藤拡二[††] 深川正一[††]
中川八穂子[††] 五百木伸洋[†††]

科学技術計算をターゲットとするスーパーテクニカルサーバ SR11000 モデル J1 を開発した．POWER5 を 16CPU 搭載する SR11000 モデル J1 のノードは，理論ピーク演算性能 121.6GFLOPS を有し，協調型マイクロプロセッサ（COMPAS）と呼ぶノード内並列処理方式と，擬似ベクトル処理（PVP）によるメモリアクセスを含めたパイプライン処理により，単一の高性能なプロセッシングエレメントとして利用できる．本稿では，COMPAS と PVP を可能とする SR11000 モデル J1 のノードアーキテクチャを紹介するとともに，ノード性能の評価結果について述べる．

Node Architecture and Performance Evaluation of the Hitachi Super Technical Server SR11000 Model J1

Hidetaka Aoki,[†] Tomohiro Nakamura,[†] Naonobu Sukegawa,[†]
Koji Saito,[††] Masakazu Fukagawa,[††] Yaoko Nakagawa[††]
and Nobuhiro Ioki[†††]

We developed the Hitachi Super Technical Server SR11000 model J1, which is suitable for scientific and technical computing.The node of SR11000 model J1, which is an SMP with 16 POWER5 CPUs with a theoretical peak performance of 121.6 GFLOPS, is designed for efficient execution of COMPAS (CO-operative Micro-Processors in single Address Space) parallel processing and PVP (Pseudo Vector Processing).This paper describes the node architecture of SR11000 model J1 and the results of performance evaluation.

1. はじめに

2004 年 10 月に製品発表した SR11000 モデル J1 は，ベクトル・スカラ融合型サーバ SR8000 シリーズの後継機として開発を進めてきたスーパーテクニカルサーバ SR11000 シリーズの第 2 世代モデルである．SR11000 モデル J1 は，1.9 GHz 動作の POWER5[1),2)] を 16CPU 搭載するメモリ共有ノードを基本構成単位とし，多段クロスバネットワークにより複数ノードを接続する構成をとる．最大 512 ノードを接続することで理論ピーク演算性能 62.2TFLOPS を実現し，大規模な科学技術計算に対応可能である．

† 株式会社日立製作所中央研究所
Central Research Laboratory, Hitachi, Ltd.
†† 株式会社日立製作所エンタープライズサーバ事業部
Enterprize Server Division, Hitachi, Ltd.
††† 株式会社日立製作所ソフトウェア事業部
Software Division, Hitachi, Ltd.

先代機である SR8000 シリーズ[3)] は，協調型マイクロプロセッサ（CO-operative Micro-Processors in single Address Space，COMPAS）と呼ぶノード内並列処理方式を採用し，さらに，擬似ベクトル処理（Pseudo Vector Processing，PVP）によるメモリアクセスを含めた高度なパイプライン処理を実現していた．これらにより，複数の RISC CPU からなるノードを，高い演算性能と高いメモリアクセス性能を持つ単一のプロセッシングエレメントとして利用することを可能にした．さらに，複数ノードを多次元クロスバネットワークにより接続することで，大規模計算における高性能・高スケーラビリティを実現した．

SR11000 モデル J1 は，SR8000 シリーズの持つこれらの特長を継承しつつ，さらなる性能向上を実現するものとして開発を進めてきた．すなわち，高速クロックレートで動作する POWER5 の採用と搭載 CPU 数の 2 倍化，および大容量キャッシュの搭載によりノード性能を大きく向上させるとともに，COMPAS や PVP

表 1 ハードウェア仕様比較
Table 1 Comparison of hardware specifications.

		SR8000 model G1	SR11000 model J1	Scaling Factor
CPU & Cache	CPU	64-bit PowerPC based 0.45 GHz	POWER5 1.9 GHz	-
	CPU Performance	1.8GFLOPS	7.6GFLOPS	4.2
	On-chip Cache	DL1: 128 KB / 1CPU	DL1: 32 KB / 1CPU L2: 1.875 MB / 2CPUs	7.5
	Off-chip Cache	-	L3: 36 MB / 2CPUs	(144)
Node	Number of CPUs	8	16	2.0
	Node Performance	14.4GFLOPS	121.6GFLOPS	8.4
	Maximum Memory	16 GB	128 GB	8.0
System	Number of Nodes	4 – 512	4 – 512	1.0
	System Performance	57.6GFLOPS – 7.3TFLOPS	486.4GFLOPS – 62.2TFLOPS	8.4
	Maximum Memory	8TB	64TB	8.0
	Network Bandwidth	1.6 GB/s × bi-direction	4/8/12 GB/s × bi-direction	7.5
	Network Topology	multi-dimensional crossbar	multi-stage crossbar	-

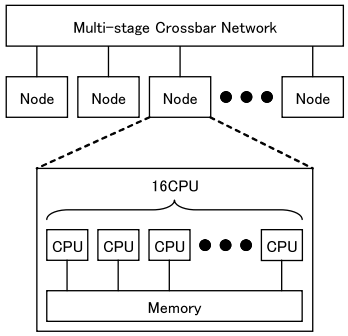

図 1 SR11000 モデル J1 のシステム構成
Fig. 1 System overview of SR11000 model J1.

を高性能で実現可能とするノードアーキテクチャを採用した☆．また，ノード性能の向上に合わせてノード間転送性能も向上させることで，複数ノード利用時の高性能・高スケーラビリティを確保した．

本稿では，SR11000 モデル J1 の設計思想の基となるプログラム実行モデルと，それを実現するノードアーキテクチャについて説明する．さらに，ノード性能を評価し，ノードアーキテクチャの有効性を確認する．以下，2 章では，SR11000 モデル J1 の概要と，想定するプログラム実行モデルについて述べる．続く 3 章で，そのプログラム実行モデルを可能とする SR11000 モデル J1 のノードアーキテクチャについて述べ，4 章でノードの性能を評価する．5 章で関連研究について述べ，最後に 6 章で本稿をまとめる．

2. SR11000 モデル J1 の概要

2.1 システム構成

SR11000 モデル J1 は，SMP（Symmetric Multi Processor）ノードをネットワークで接続した並列コンピュータであり，大規模科学技術計算を主なターゲットとしている．SR11000 モデル J1 のシステム構成を**図 1** に示す．また，先代機 SR8000 モデル G1 とのハードウェア仕様の比較を**表 1** に示す．

SR11000 モデル J1 は，IBM と共同開発した 1.9 GHz 動作の POWER5 プロセッサを 1 ノードに 16CPU 搭載し，SR8000 モデル G1 の 8.4 倍となるノード理論ピーク演算性能 121.6GFLOPS を実現した．また，高い実効性能を実現するために，POWER5 プロセッサに搭載されたオンチップ L2 キャッシュに加え，L2 キャッシュのビクティムキャッシュとして機能する大容量のオフチップ L3 キャッシュを搭載した．搭載可能なメモリ容量は，ノード演算性能の向上に合わせ，SR8000 モデル G1 の 8 倍となるノードあたり最大 128 GB に拡大した．

ノード間接続には，転送データの衝突が少ない多段クロスバネットワークを採用した．アプリケーションプログラムの特性に合わせて構成を変えられるマルチリンク構成のネットワークであり，ノード間転送性能は最大 12 GB/s（単方向）× 2 である．ノード演算性能の向上に合わせてノード間転送性能も約 8 倍に向上することで，SR8000 モデル G1 とほぼ同等の性能バランス（ノード間転送性能対ノード演算性能）を達成し，複数ノードにまたがる大規模計算における高いスケーラビリティを実現した．

また，SR11000 モデル J1 はコンパクトな実装を特長とし，メインフレームで培った高密度実装技術により，単位面積あたりの理論ピーク演算性能は約 640GFLOPS/m^2 と，世界最高クラスを達成した．これにより，小さな設置面積で数 TFLOPS クラスのシ

☆ 本稿では，ノードの構成方式に加え，ノード性能を向上させる種々の機構や手法を含めて，ノードアーキテクチャと呼ぶ．

表 2 SR11000 モデル J1 のプログラム実行モデル
Table 2 Program processing model of SR11000 model J1.

階層	プログラム実行モデル
ノード間並列化	MPI
ノード内並列化	COMPAS
CPU 処理高速化	PVP + 大容量キャッシュ活用

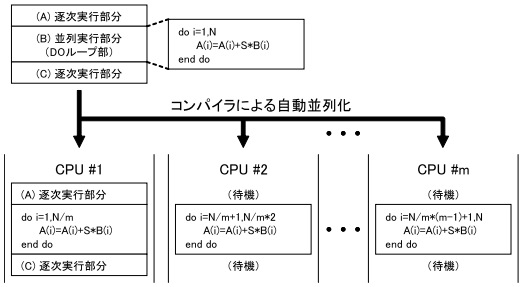

図 2 COMPAS 実行モデル
Fig. 2 Processing model of COMPAS.

ステムを導入可能である．

2.2 プログラム実行モデル

SR8000 シリーズでは，主要なプログラム実行モデルとして，(1) MPI によるノード間並列化，(2) COMPAS によるノード内並列化，および，(3) PVP による CPU 処理の高速化の 3 階層を規定している．SR11000 モデル J1 では，SR8000 シリーズのプログラム実行モデルを継承しつつ，大容量キャッシュによる CPU 処理のさらなる高速化を目指した．SR11000 モデル J1 で想定するプログラム実行モデルを，**表 2** にまとめる．以下，特長技術である COMPAS と PVP について説明する．

(1) **COMPAS**

SR11000 モデル J1 では，大規模科学技術計算における高いデータ並列性を利用し，COMPAS と呼ぶノード内並列処理方式を採用した．COMPAS では，自動並列化コンパイラにより逐次実行部分と並列実行部分から構成される SIMD 並列コードを生成し，これをノード内の複数 CPU で実行する．

COMPAS において，DO ループを m 個の CPU で並列処理する際の実行モデルを，**図 2** に示す．COMPAS により，ユーザはアルゴリズムレベルからプログラムを並列化することなく，ノード内複数 CPU の並列処理による高い性能を享受できる．SR11000 モデル J1 では，ノード内のすべての CPU による COMPAS 実行に加え，ノードを分割し，同時に複数のプログラムを COMPAS 実行することが可能である．

COMPAS では，逐次実行部分と並列実行部分との間，および，並列実行部分どうしの間で，データの同期をとるためのバリア同期を行う．バリア同期をはさむことにより，複数 CPU から発行された同一アドレスへのアクセスの順序を規定し，データの依存関係を保証する．**図 3** に，COMPAS とバリア同期の関係を示す．逐次実行部分 S_1，S_2，S_3 と並列実行部分 P_1，P_2，P_3 からなるプログラムコードの間でバリア同期を行うことで，CPU 間の実行タイミングを揃え，データの分配・収集・交換を可能にする．

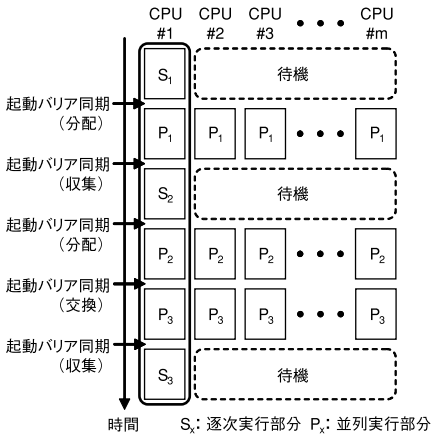

図 3 COMPAS とバリア同期
Fig. 3 COMPAS and barrier synchronization.

(2) **PVP**

SR11000 モデル J1 は大容量キャッシュを搭載するが，科学技術計算ではキャッシュに入りきらない大規模データを扱うことも多く，幅広いアプリケーションで高性能を実現するためには，メモリ上の大規模データへのアクセス性能を高めることが必要である．SR11000 モデル J1 では，この要求に対応して PVP を採用した．

PVP では，メモリからデータをパイプライン的に取り込むことで，メモリレイテンシを隠蔽し，メモリ上の大規模データを演算器へ高速に供給する．SR11000 モデル J1 では，キャッシュへのデータプリフェッチをパイプライン的に行うことで PVP を実現し，ベクトル型スーパーコンピュータと同様な大規模データの高速処理を実現する．具体的には **図 4** に示すように，ロード命令によるレジスタへのデータ読み込みに先立ち，メモリからキャッシュにデータをプリフェッチする．プリフェッチによるデータ転送はキャッシュライン単位で行われ，**図 5** に示すように，1 キャッシュライン分の処理（図では load，add，store）に先行してプリフェッチ要求を発行することで，メモリからキャッシュへのデータ転送を含めたパイプライン処理を実現する．

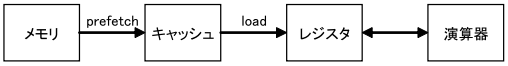

図 4 プリフェッチによる PVP 実行モデル
Fig. 4 Processing model of prefetch-based PVP.

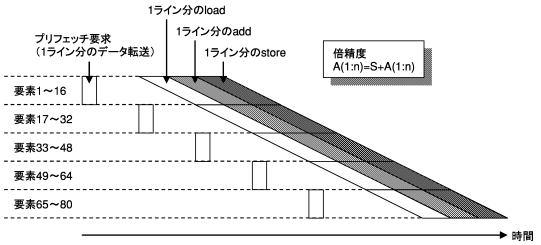

図 5 プリフェッチによる PVP のパイプライン処理
Fig. 5 Pipelined processing with prefetch-based PVP.

2.3 ノードアーキテクチャにおける課題

2.2 節に述べた COMPAS と PVP により高性能を達成するには，ノードで以下の特性を実現することが課題となる．

(1) **メモリ資源の均一性と高いメモリアクセス性能**

一般に，各 CPU のアクセスするデータがローカライズされている場合には，そのデータを各 CPU に物理的・論理的に近いメモリに配置することで，高速なデータアクセスが可能となる．しかし，このようなローカライズには，アルゴリズムレベルからの並列実行最適化が必須である．コンパイラによりプログラムの自動並列化を行う COMPAS では，メモリ上のデータ配置を考慮した並列化は難しく，また，このような並列化を前提とすることは自動並列化の適用範囲を著しく狭めることになる．COMPAS を採用する SR11000 モデル J1 では，メモリ上のデータ配置を意識せずに処理を並列化した場合にもメモリバンド幅やメモリレイテンシの悪化による性能低下が発生しないよう，ノード内のメモリアクセスの均一性を高める必要がある．

高性能な PVP を実現するうえでも，高いメモリアクセス性能が重要である．特に，COMPAS と PVP の併用により，並列実行部分で複数 CPU が時間的に集中してメモリアクセスを行うことになるため，上記メモリ資源の均一性確保と合わせ，ノード内の全 CPU がノード内の全メモリ資源にアクセスするケースでの高性能が必要となる．

(2) **プリフェッチによる安定したデータ供給**

利用が見込まれるデータをあらかじめキャッシュに登録するプリフェッチ方式には，ハードウェアプリフェッチとソフトウェアプリフェッチとがある[4]．ハードウェアプリフェッチとは，規則的なデータアクセスパターンを実行時に検出し，ハードウェアで自動的にプリフェッチを行う方式である．これに対しソフトウェアプリフェッチは，コンパイル時にデータアクセスパターンを解析し，ロードモジュール中にプリフェッチ命令を挿入することで実現する．

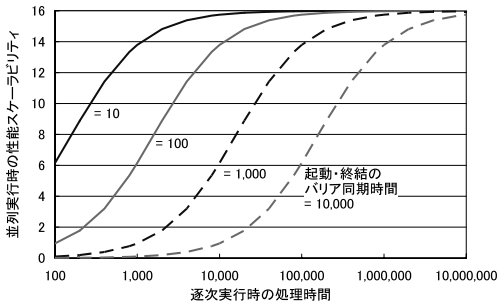

図 6 バリア同期時間と並列実行時の性能スケーラビリティ
Fig. 6 Barrier synchronization time and parallel processing scalability.

POWER5 はハードウェアプリフェッチ機構を搭載し，ロード命令による読み出しが予想されるデータをプリフェッチできる．しかし，一般にハードウェアプリフェッチでは，ハードウェア資源の制約により，プリフェッチ可能なストリーム数に上限があり，ストリーム数が上限を超えると性能低下が発生する．

プリフェッチによる PVP で安定した高性能を実現するためには，ストリーム数によらない安定した高い実効メモリバンド幅の実現が必要である．

(3) **高速なバリア同期**

並列処理において，使用する CPU 数に応じた高い性能スケーラビリティを得るには，逐次実行部分の処理時間（逐次実行時間）が並列実行部分の処理時間（並列実行時間）に比べて十分小さいこととともに，バリア同期に要する時間が並列実行時間に比べて十分小さいことが必要である．

特に，並列処理部分の処理量が少ない場合には，並列処理の起動・終結などによるバリア同期時間の影響が大きくなる．**図 6** に，DO ループを並列処理する場合の，バリア同期時間と並列スケーラビリティの関係を示す．グラフの横軸は，DO ループを 1CPU で逐次実行した場合の処理時間を表す．また，縦軸は逐次実行時の性能を 1 とした相対性能であり，並列実行による性能スケーラビリティを表す．起動・終結のバリア同期時間に応じた，16CPU 並列実行時の理論性能をプロットした．DO ループの処理量が少ない場合にも COMPAS による並列処理で性能向上を実現するには，バリア同期時間を低減させることが重要となる．

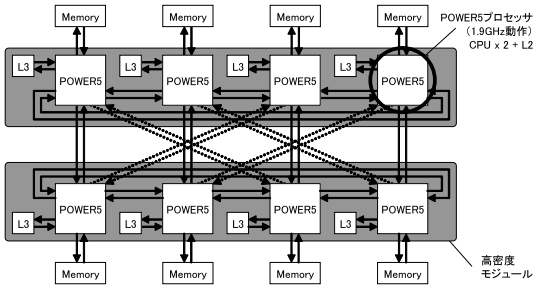

図 7 SR11000 モデル J1 のノード構成
Fig. 7 Node architecture of SR11000 model J1.

3. ノードアーキテクチャ

2.3 節にあげた課題を解決し，高性能な COMPAS と PVP を実現するよう，SR11000 モデル J1 のノードアーキテクチャを決定した．

SR11000 モデル J1 のノード構成を **図 7** に示す．2CPU コアを持つ POWER5 プロセッサ 4 個と L3 キャッシュを高密度モジュール（High Density Module，HDM）に実装したうえ，2 つの高密度モジュール間を 8 組のデータ線で密に結合する構成を採用した．

3.1 メモリ資源の均一性確保と高いメモリアクセス性能の実現

SR11000 モデル J1 のノードでは，メモリ資源の均一性を確保し，かつ，高いメモリアクセス性能を実現するため，以下の技術を導入した．

(1) **高速メモリ素子**

メモリ素子として，データ転送性能の高い DDR2-SDRAM を採用し，メモリアクセスにおける高いバンド幅を確保した．

(2) **フラットメモリインタリーブ**

COMPAS による並列処理においてメモリアクセス性能を高めるには，各 CPU と接続される合計 8 個のメモリ資源を全 CPU が均等に使用することで，一部メモリ資源へのアクセス集中に起因する性能低下を抑止する必要がある．SR11000 モデル J1 では，128 B のキャッシュライン単位でデータを全メモリ資源に分散させるフラットメモリインタリーブを採用し，特定メモリ資源へのアクセスの集中を回避する．これにより，データの配置によらない安定した高いメモリアクセス性能を実現する．

(3) **HDM 間強結合**

フラットメモリインタリーブによりデータが全メモリ資源に均等に分散配置された状態では，メモリアクセスによるデータ転送の半分が HDM 間インタフェースを通る．これは，各 CPU のメモリアクセスの半分はローカル HDM（自 CPU の属する HDM）のメモリに，残りの半分はリモート HDM（自 CPU の属さない HDM）のメモリに対してなされるためである．

POWER5 は，8 チップ/16CPU のプロセッサ・ブックを増設単位とし，最大 64CPU の SMP ノードを付加 LSI なしで構成できる．プロセッサ・ブックは，図 7 において点線で示した HDM 間のデータ線を除いた構成であり，HDM 間は実線で示した 4 組のデータ線で接続される[5]．

これに対し SR11000 モデル J1 では，16CPU 構成までであることを利用し，点線のデータ線を加えた 8 組のデータ線で HDM 間を接続した．これを，HDM 間強結合と呼ぶ．データ線を 2 倍に増強することで HDM 間インタフェースのバンド幅ボトルネックを解消するとともに，点線で示したデータ線を用いることでデータ転送のホップ数を削減☆してレイテンシを短縮し，メモリ資源の均一性を高めた．

3.2 プリフェッチによる安定したデータ供給

SR11000 モデル J1 では，PVP 実現のため，キャッシュへのデータプリフェッチをパイプライン的に行う．データプリフェッチ方式としては，POWER5 によるハードウェアプリフェッチに加え，ソフトウェアプリフェッチの一種であるソフトウェアアシストプリフェッチ[6] を併用する．

POWER5 のハードウェアプリフェッチでは，連続するキャッシュラインへの読み出しアクセス（昇順または降順）からロードストリームを検出し，L1 キャッシュおよび L2 キャッシュに対して階層的なプリフェッチを行う[4]．定常状態では，ロード命令が L1 キャッシュ中のあるキャッシュラインに初めてアクセスするのを契機として後続キャッシュラインをプリフェッチするため，ハードウェアプリフェッチにより，図 5 に示した PVP におけるプリフェッチ要求は，非明示的に発行されることになる．POWER5 では，各 CPU コアが最大 8 個のロードストリームを検出し，ハードウェアプリフェッチを行う．

しかし，8 個を超えるロードストリームを含むループでは，すべてのストリームをハードウェアプリフェッチすることができないため，性能が低下する．この問題を解消するため，SR11000 モデル J1 ではソフトウェアアシストプリフェッチを導入した．

PowerPC アーキテクチャには，dcbt（Data Cache Block Touch）命令というプリフェッチ命令が用意され

☆ データ転送における CPU ホップ数の最大値は，HDM 間強結合なしの 3 ホップに対し，HDM 間強結合導入により 2 ホップに削減．

表 3 プリフェッチ方式の比較
Table 3 Comparison of prefetch methods.

	ハードウェアプリフェッチ	ソフトウェアアシストプリフェッチ
連続アクセス	ハードウェア検出可能	コンパイラ検出可能
インデックスアクセス	ハードウェア検出可能	コンパイラ検出不可（ユーザ指定要）
対応可能ストリーム数	1〜8 ストリーム	制限なし

```
do i=1,m
   S=S+A1(i)+A2(i)+...+An(i)
end do
```

a) 適用前のコード

```
do i=1,m-U+1,U
   dcbt(TH=0001) for A1(i+AHEAD)
   dcbt(TH=0001) for A2(i+AHEAD)
   ...
   dcbt(TH=0001) for An(i+AHEAD)
   S=S+A1(i)+A2(i)+...+An(i)
   S=S+A1(i+1)+A2(i+1)+...+An(i+1)
   ...
   S=S+A1(i+U-1)+A2(i+U-1)+...+An(i+U-1)
end do
do i=i,m
   S=S+A1(i)+A2(i)+...+An(i)
end do
```

b) 適用後のコードイメージ

図 8 ソフトウェアアシストプリフェッチの適用例
Fig. 8 Sample code of software-assisted prefetch.

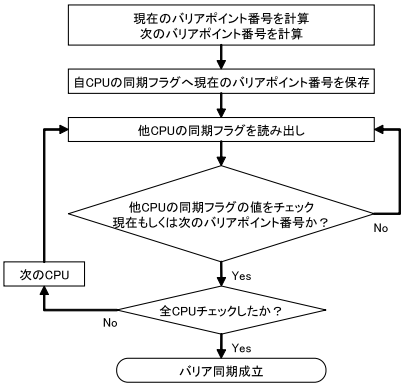

図 9 FBS 方式の処理フロー
Fig. 9 Flowchart of FBS method.

ている．POWER5 ではこの dcbt 命令が拡張されており，TH（Touch Hint）フィールドに 0001 または 0011 を指定することで，指定アドレスをプリフェッチするだけでなく，昇順または降順のロードストリームの存在をヒントとしてあたえることができる[7]．SR11000 モデル J1 で採用するソフトウェアアシストプリフェッチでは，ループ中の全ロードストリームに対してコンパイラが TH=0001 または 0011 を指定した dcbt 命令を発行することで，ソフトウェアによりロードストリームに関するヒントをあたえ，ハードウェアプリフェッチにおけるストリーム数の上限を解消する．dcbt 命令は，各ストリームに対し，少なくとも 1 キャッシュラインに 1 回の割合で発行する．

図 8 に，ソフトウェアアシストプリフェッチの適用例を示す．ここで AHEAD（≥ 0）とは，ロード命令に先行して dcbt 命令を発行するためのパラメータである．また，ループを U 倍（倍精度データでは $U \leq 16$）に展開することで，同一キャッシュラインに発行される dcbt 命令を 1/U に削減する．SR11000 向けの日立最適化 FORTRAN90 コンパイラでは，本手法の自動適用をサポートしている．

ハードウェアプリフェッチとソフトウェアアシストプリフェッチの特性を **表 3** にまとめる．SR11000 モデル J1 では，これら 2 種類のプリフェッチを併用することで，PVP における安定した高いメモリアクセス性能を実現した．

3.3 高速なバリア同期

バリア同期時間を低減させるため，SR11000 モデル J1 では，BSR-FBS（Barrier Synchronization Register - Fast Barrier Synchronization）と呼ぶバリア同期処理方式を採用した．BSR-FBS 方式では，ロック変数（mutex lock）による排他処理を不要とした高速なソフトウェアバリア同期処理方式である FBS 方式[8] をベースに，BSR と呼ぶハードウェアにより，さらなる高速化を実現する．

FBS 方式では，通常のロード・ストア命令などを用いてバリア同期を実現する．FBS 方式の処理フローを **図 9** に示す．FBS 方式では，それぞれの CPU が同期フラグを持ち，同期フラグには各 CPU の到達したバリアポイント番号を保持する☆．そして，この値が並列処理に参加している CPU 間で一定の条件に達するのを待つことで，バリア同期を実現する．一定の条件とは，他の全 CPU のバリアポイント番号が，自 CPU のバリアポイント番号と同一か，その次のバリアポイント番号になることである．この処理ではロッ

☆ プログラム中のバリア同期処理を行う部分をバリアポイントと呼び，複数のバリアポイントを区別するための番号をバリアポイント番号と呼ぶ．

表 4 COMPAS 実現方式の比較
Table 4 Comparison of COMPAS implementations.

	COMPAS 実行単位	バリア同期
SR8000 model G1	8CPU/14.4GFLOPS	専用命令 + 専用ハードウェア
SR11000 model J1	16CPU/121.6GFLOPS （ノード分割による複数実行可）	BSR-FBS 方式 （ソフトウェア処理 + 高速化ハードウェア）

表 5 PVP 実現方式の比較
Table 5 Comparison of PVP implementations.

	ハードウェア	ソフトウェア
SR8000 model G1	No	Yes（プリフェッチ命令，プリロード命令）
SR11000 model J1	Yes（ハードウェアプリフェッチ）	Yes（プリフェッチ命令）

ク変数による排他処理は不要で，各 CPU は他の CPU の同期フラグを監視することで同期待ちを行う．

BSR-FBS 方式では，FBS 方式における同期フラグを，BSR と呼ぶメモリマップされたハードウェア資源に格納する．BSR はストアされた値をノード内の全 CPU に高速に伝達する機能を持ち，これにより FBS 方式のさらなる高速化を実現した．

3.4 SR8000 シリーズからの継続性

本節では，SR11000 モデル J1 で実現される COMPAS および PVP を，SR8000 シリーズと比較する．

COMPAS 実現方式における SR8000 シリーズとの違いを，**表 4** に示す．SR11000 モデル J1 では，COMPAS の実行単位となるノード性能が大幅に向上している．SR8000 シリーズから移行する際のノード内 CPU 数の違いは，コンパイラによる自動並列化により吸収できる．また，8CPU による並列実行が不可欠なケースでも，SR11000 モデル J1 のノードを 8CPU ごとに 2 分割することで対応可能である．SR8000 シリーズでは専用命令と専用ハードウェアにより高速なバリア同期を実現していたが，SR11000 モデル J1 では BSR-FBS 方式（ソフトウェア処理と高速化ハードウェア）により実現した．

SR8000 シリーズとの PVP 実現方式の違いを，**表 5** に示す．SR11000 モデル J1 ではハードウェアプリフェッチを導入したことで，コンパイラでは検出できなかったインデックスアクセスの性能を高めた．SR8000 シリーズがサポートしていたプリロード命令（キャッシュを使わずに，データをレジスタに直接ロードする方式）は，キャッシュの大容量化にともない，SR11000 モデル J1 ではサポートしない．

4. ノード性能の評価

本章では，3 章で述べたアーキテクチャを有する SR11000 モデル J1 のノード性能を評価する．以下の評価では，日立最適化 FORTRAN90 コンパイラを用いてロードモジュールを作成し，コンパイルオプションとしては `-64 -model=J1 -Oss` を適用した．4.3 節の評価ではさらに，コンパイラ内部パラメータによりソフトウェアアシストプリフェッチ適用の有無を指示した．いずれの場合も，ループ分割をせずに単一ループとして処理を行い，かつ，ループ長 m で並列化するコードが生成される．プログラム実行にあたっては，SR11000 モデル J1 の 1 ノードを用い，ページサイズを 16MB とするラージページモードで行った．

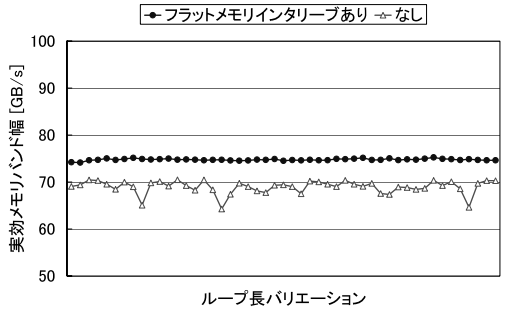

図 10 DAXPY における実効メモリバンド幅
Fig. 10 Sustained memory bandwidth in DAXPY.

4.1 フラットメモリインタリーブの効果

DAXPY カーネル `A(1:m)=A(1:m)+S*B(1:m)` を用いて，フラットメモリインタリーブの効果を評価した．ループ長 m は十分に大きく，すべてのデータ読み出しがキャッシュミスする．フラットメモリインタリーブを有効にした場合と無効にした場合の実効メモリバンド幅を，同一のロードモジュールを用いて測定した．ループ長 m を変化させて 16CPU で COMPAS 実行した結果を，**図 10** に示す．グラフ横軸はループ長 m のバリエーションであり，アクセスするデータのアドレスバリエーションに相当する．

フラットメモリインタリーブを無効にした場合では，オペレーティングシステム AIX 5L の仮想アドレス－実アドレス変換を利用したページ単位のインタリーブを行っている．ページ単位のインタリーブでは，両

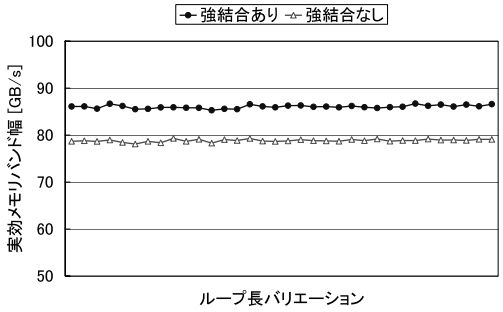

図 11 4 配列総和における実効メモリバンド幅
Fig. 11 Sustained memory bandwidth in sum of 4 arrays.

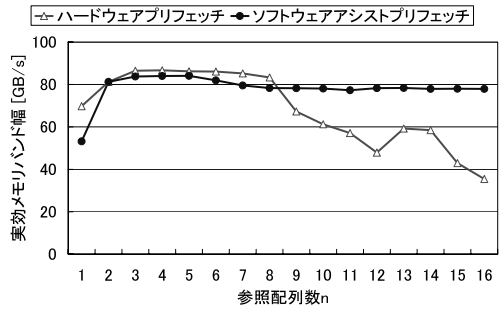

図 12 n 配列総和における実効メモリバンド幅
Fig. 12 Sustained memory bandwidth in sum of N arrays.

HDM に 1 ページごとにデータがインタリーブされ，各ページは HDM 内の 4 個のメモリ資源で 128 B 単位にインタリーブされる．測定は 16 MB のラージページ環境下で行ったため，フラットメモリインタリーブを無効にした場合には，HDM 間は 16 MB 単位でデータがインタリーブされる．データ全体で見ると全メモリ資源にほぼ均等に分散配置されるが，ある短い時間で見ると，各 CPU がアクセスするデータの配置に HDM 間の偏りが生じやすくなる．実効メモリバンド幅はデータ配置の影響を受けやすく，ループ長によって最大 10%の性能ばらつきが生じている．

これに対し，フラットメモリインタリーブを有効にした場合には，HDM 間は 512 B（= 128 B × 4）単位でデータがインタリーブされる．HDM 間のインタリーブ単位を細粒度にすることで，短い時間で見ても 8 個のメモリ資源を均等に利用可能となる．フラットメモリインタリーブを有効にすることで，ループ長によらず安定して高い実効メモリバンド幅（平均で+8%）を実現している．

4.2 HDM 間強結合の効果

4 個の倍精度配列データの総和を求めるコード `S=A1(1:m)+A2(1:m)+A3(1:m)+A4(1:m)` を用いて，HDM 間強結合の効果を測定した．ループ長 m は十分に大きく，すべてのデータ読み出しがキャッシュミスする．HDM 間強結合を有効にした場合と無効にした場合の実効メモリバンド幅を，同一のロードモジュールを用いて測定した．ループ長 m を変化させて 16CPU で COMPAS 実行した結果を，**図 11** に示す．HDM 間強結合により，ループ長によらず実効メモリバンド幅が向上し，平均で+9%の性能向上となっている．

なお，CPU 間でキャッシュ内データの転送を繰り返すテストプログラムでは，HDM 間の強結合により最大で+30%以上の性能向上を実測している．

4.3 プリフェッチの性能

n 個の倍精度配列データの総和を求めるコード `S=A1(1:m)+A2(1:m)+...+An(1:m)` を用いて，ハードウェアプリフェッチとソフトウェアアシストプリフェッチの特性を評価した．ループ長 m は十分に大きく，すべてのデータ読み出しがキャッシュミスする．参照配列数 n を 1 から 16 まで変化させ 16CPU で COMPAS 実行した場合の実効メモリバンド幅を，**図 12** に示す．

ハードウェアプリフェッチ，ソフトウェアアシストプリフェッチとも，ロードストリーム数（=参照配列数）が 1 の場合に性能が低いのは，プリフェッチを行うストリーム数が少なく，メモリレイテンシを十分に隠蔽できないためである．ハードウェアプリフェッチでは，ロードストリーム数が 8 を超えると性能が大きく低下する．これに対し，ソフトウェアアシストプリフェッチを適用することで，ロードストリーム数が増えた場合でもピークの約 9 割の高い性能を安定して実現できる．

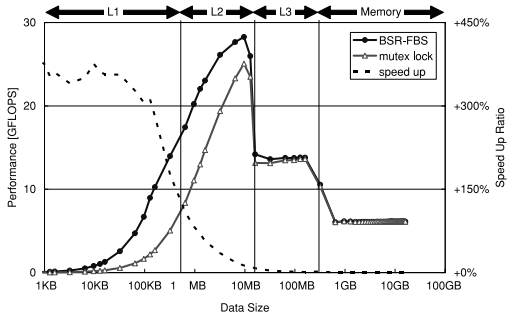

図 13 DAXPY 性能
Fig. 13 DAXPY performance.

4.4 BSR-FBS 方式の効果

DAXPY カーネル `A(1:m)=A(1:m)+S*B(1:m)` を用いて，COMPAS 実行における BSR-FBS 方式の効果を評価した．**図 13** は，DAXPY におけるデータサイズと 16CPU による COMPAS 実行時の性能の関係を示したグラフである．BSR-FBS 方式，ロック方式（mutex lock）それぞれを使用した場合の演算性能に

加え，ロック方式に対する BSR-FBS 方式の性能向上率を示した．

BSR-FBS 方式の採用により，特に L2 キャッシュ領域までで顕著な性能向上が見られ，ロック方式と比べて，データサイズ 1 MB 付近で+80%以上，10 MB 付近で+15%以上の性能向上を得られる．

5. 関連研究

Blue Planet[9)~11)] では，1 コアのみ搭載する POWER5 プロセッサ 8 個で構成された 8CPU ノードを用い，ViVA（Virtual Vector Architecture）と呼ぶ並列実行モデルにより 1 ノードを単体の高速なプロセッシングエレメントとして動作させる．BSR による高速なバリア同期を利用して ViVA を実現するとされるが，現時点では詳細は公表されていない．

Blue Planet では，CPU あたりのキャッシュ容量やメモリバンド幅を強化するために，POWER5 プロセッサあたりの CPU コア数を半分に減らすが（2 → 1），これによりノードの理論ピーク演算性能は半減する．これに対し SR11000 モデル J1 は，Blue Planet と同数の POWER5 プロセッサを搭載しながら，2CPU コアを持つ POWER5 プロセッサにより高い理論ピーク演算性能を確保するとともに，本稿で述べたノードアーキテクチャの工夫により高いメモリアクセス性能を実現する．

また，Blue Planet では，高速なデータアクセスを実現するためにデータを各 CPU にローカライズする．これに対し SR11000 モデル J1 では，ノード内のメモリ資源の均一性を高めることで，自動並列化の適用範囲を広げるアプローチを採る．

6. おわりに

SR8000 シリーズの持つ特長を継承し，さらなる性能向上を実現する，SR11000 モデル J1 を開発した．POWER5 を 16CPU 搭載する SR11000 モデル J1 のノードは，理論ピーク演算性能 121.6GFLOPS を有し，協調型マイクロプロセッサ（COMPAS）と呼ぶノード内並列処理方式と，擬似ベクトル処理（PVP）によるメモリアクセスを含めたパイプライン処理により，単一の高性能なプロセッシングエレメントとして利用できる．

本稿では，これらを可能としたノードアーキテクチャについて説明した．SR1100 モデル J1 では，フラットメモリインタリーブや HDM 間強結合によるメモリ資源の均一性確保と高いメモリアクセス性能の実現，プリフェッチによる安定したデータ供給，および，高速なバリア同期を実現する BSR-FBS 方式の採用により，高性能な COMPAS および PVP を実現した．本稿ではさらに，ノード性能を評価し，開発したノードアーキテクチャの有効性を示した．優れたノードアーキテクチャとその性能を引き出す最適化コンパイラ[12)] により，SR11000 モデル J1 は，多様なユーザアプリケーションにおいて高い実効性能を発揮する．

参考文献

1) Adra, B., Blank, A., Gieparda, M., Haust, J., Stadler, O. and Szerdi, D.: *Advanced POWER Virtualization on IBM eServer p5 Servers: Introduction and Basic Configuration*, 1st edition, IBM Redbooks, SG24-7940-00 (2004).
2) Gibbs, B., Atyam, B., Berres, F., Blanchard, B., Castillo, L., Coelho, P., Guerin, N., Liu, L., Maciel, C.D., Thirumalai, R. and Sosa, C.: *Advanced POWER Virtualization on IBM eServer p5 Servers: Architecture and Performance Considerations*, IBM Draft Redbooks, SG24-5768-00 (2004).
3) Tamaki, Y., Sukegawa, N., Ito, M., Tanaka, Y., Fukagawa, M., Sumimoto, T. and Ioki, N.: Node Architecture and Performance Evaluation of the Hitachi Super Technical Server SR8000, *Proc.12th International Conference on Parallel and Distributed Computing Systems*, pp.487–493 (1999).
4) Wang, Z., McKinley, K.S. and Burger, D.: Combining Cooperative Software/Hardware Prefetching and Cache Replacement, *5th Annual Austin CAS Conference* (2004).
5) Domberg, P., Kelley, N., Kim, T. and Wei, D.: *IBM eServer p5 590 and 595 System Handbook*, 1st edition, IBM Redbooks, SG24-9119-00 (2005).
6) 青木秀貴，處 雅尋，本川敬子，五百木伸洋，齋藤拡二：SR11000 におけるソフトウェアプリフェッチ手法の評価，情報処理学会研究報告，2004-ARC-159, pp.109–114 (2004).
7) IBM: *AIX 5L Version 5.3 Assembler Language Reference*, 1st edition, SC23-4923-00 (2004).
8) 中村友洋，高山恒一，青木秀貴，松居昭宏，助川直伸：SR11000 モデル H1 におけるバリア同期の高速化手法，情報処理学会研究報告，2003-ARC-155, pp.69–74 (2003).
9) McCurdy, C.W., Stevens, R., Simon, H., Kramer, W., Bailey, D., Johnston, W., Catlett, C., Lusk, R., Morgan, T., Meza, J., Banda, M., Leighton, J. and Hules, J.: Creating Science-Driven Computer Architecture: A New Path

to Scientific Leadership, Lawrence Berkeley National Laboratory Report LBNL/PUB-5483 (2002).
10) Kramer, B.: Creating Science-Driven Computer Architecture: Blue Planet, *ScicomP 7* (2003).
11) Simon, H., Kramer, W., Saphir, W., Shalf, J., Bailey, D., Oliker, L., Banda, M., McCurdy, C. W., Hules, J., Canning, A., Day, M., Colella, P., Serafini, D., Wehner, M. and Nugent, P.: Science-Driven System Architecture: A New Process for Leadership Class Computing, *Journal of the Earth Simulator*, Vol.2, pp.2–10 (2005).
12) 橋本博幸，本川敬子，久島伊知郎：SR11000 向け実行資源均等化命令スケジューリング，情報処理学会研究報告，2005-ARC-161, pp.27–32 (2005).

(平成 17 年 1 月 24 日受付)
(平成 17 年 4 月 18 日採録)

青木　秀貴（正会員）
1972 年生．1997 年京都大学大学院工学研究科情報工学専攻修士課程修了．同年（株）日立製作所入社．スーパーコンピュータの研究開発に従事．

中村　友洋（正会員）
1972 年生．1999 年東京大学大学院工学系研究科電気工学専攻博士課程修了．同年（株）日立製作所入社．スーパーコンピュータの研究開発に従事後，ディペンダブルシステムの研究開発に従事．工学博士．2004 年より 1 年間スタンフォード大学客員研究員．

助川　直伸
1967 年生．1992 年東京大学大学院工学系研究科電子工学専攻修士課程修了．同年（株）日立製作所入社．スーパーコンピュータの研究開発に従事．

齋藤　拡二
1962 年生．1987 年東京大学大学院工学系研究科電気工学専攻修士課程修了．同年（株）日立製作所入社．大形汎用機，RISC プロセッサ，Itanium サーバ等の開発を経て，現在スーパーコンピュータの開発に従事．

深川　正一
1960 年生．1985 年東京大学大学院工学系研究科電気工学専攻修士課程修了．同年（株）日立製作所入社．スーパーコンピュータ，PC クラスタの開発に従事．

中川八穂子
1957 年生．1981 年東京大学理学部情報科学科卒業．同年（株）日立製作所入社．スーパーコンピュータおよび超大形汎用機の開発に従事．「ベクトル・スカラー融合型スーパーコンピュータ SR8000 テクノロジー」で 2004 年市村産業賞貢献賞受賞．

五百木伸洋（正会員）
1963 年生．1987 年中央大学大学院理工学研究科土木工学専攻博士課程前期課程修了．同年（株）日立製作所入社．数値計算ライブラリ，コンパイラの開発に従事．日本応用数理学会会員．